》介绍

Transparent Data Encryption(TDE)可对敏感数据进行加密，如信用卡号。加密的数据存储在表或表空间中。对具有访问权限的用户透明解密。当数据文件或存储介质被盗，有助于保护数据。

我们通常使用密码验证，权限授权，数据库审计来保护数据库中的数据。但对物理结构上的数据文件没有保护，Oracle提供了TDE来加密数据文件中的敏感数据。如机密内容，身份证号码。

TDE加密密钥保存在外部安全模块中，存储在数据库外部。

》优点

从数据文件层面保护敏感数据

遵从法规或行业要求

不需要创建触发器

对用户透明，透明加密解密。用户不需要关心数据是否加密

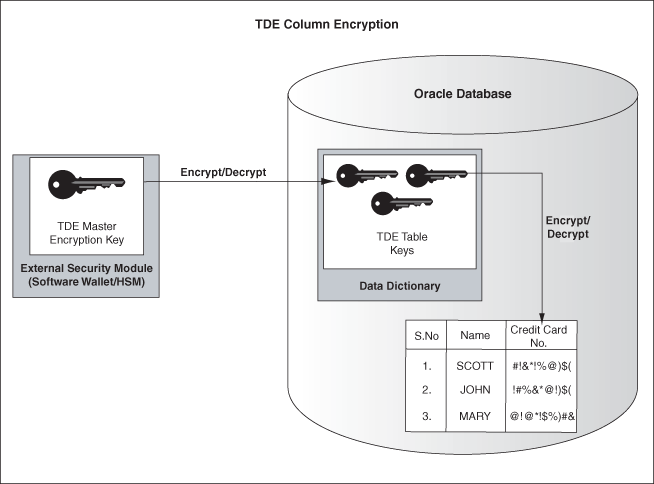
应用程序不需要对数据做操作，加密和解密都是在数据库中完成的

加密密钥管理是自动的，用户不需要管理密钥

》类型

TDE column encryption

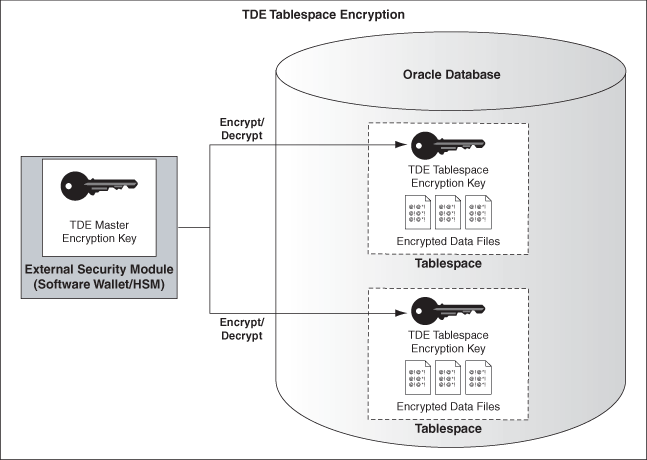
》=10.2



无论表上的加密列有多少，都只有一个表密钥

TDE tablespace encryption

》=11.1



》配置

1 使用钱包（wallet）作为外部安全模块，保存主密钥

oracle建议对TDE使用一个单独的的钱包

在sqlnet.ora中使用参数 ENCRYPTION\_WALLET\_LOCATION指定钱包的位置

ENCRYPTION\_WALLET\_LOCATION= (SOURCE=(METHOD=FILE)(METHOD\_DATA= (DIRECTORY=/u02/app/oracle/product/11.2.0/dbhome\_1/wallet)))

生成主密钥

SQL> ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY "oracle123";

打开关闭钱包

SQL> ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY "oracle123";

SQL> ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY "oracle123";

V$ENCRYPTION\_WALLET displays information on the status of the wallet and the wallet location for TDE

重置主密钥

ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY "oracle321";

数据库只有读取钱包中的主密钥才能加密解密数据

默认情况下，TDE 使用具有192位密钥长度 (AES192) 的 AES 加密算法。可以指定非默认加密算法

默认情况下，TDE使用salt选项，加密之前将随机码添加到明文，这样很难破解加密。可以使用no salt取消随机码添加

默认情况下，TDE使用SHA-1完整性算法生成Authentication Code (MAC)，做完整性检查。可以使用nomac取消完整性检查，这样会提高性能节省存储空间。

注意：可以将钱包配置为autologin方式

2 列加密

启用列加密

CREATE TABLE em (

     first\_name VARCHAR2(128),

     last\_name VARCHAR2(128),

     empid NUMBER,

     salary NUMBER(6) ENCRYPT

);

CREATE TABLE emp (

     first\_name VARCHAR2(128),

     last\_name VARCHAR2(128),

     empid NUMBER,

     salary NUMBER(6) ENCRYPT USING '3DES168' NO SALT

);

SQL> ALTER TABLE emp MODIFY (first\_name ENCRYPT);

TDE also enables you to specify a nondefault encryption algorithm. You can choose from one of the following algorithms:

3DES168

AES128

AES192 (default)

AES256

注意：在加密列上创建索引，加密列不能使用salt方式

SQL> ALTER TABLE employee MODIFY (first\_name ENCRYPT NO SALT);

create index first\_name\_indx on emp (first\_name);

取消列加密

SQL> ALTER TABLE employee MODIFY (first\_name DECRYPT);

限制很多！！

Index types other than B-tree

Range scan search through an index

External large objects (BFILE)

Synchronous Change Data Capture

Transportable Tablespaces

Original import/export utilities

In addition, you cannot use TDE column encryption to encrypt columns used in foreign key constraints.

The ALL\_ENCRYPTED\_COLUMNS view displays encryption information about encrypted columns in the tables accessible to the current user.

3 表空间加密

启用表空间加密

CREATE TABLESPACE securespace2

DATAFILE '/u02/app/oracle/secure01.dbf'

SIZE 10M

ENCRYPTION

DEFAULT STORAGE(ENCRYPT);

DBA\_TABLESPACES: The ENCRYPTED column indicates whether a tablespace is encrypted

USER\_TABLESPACES: The ENCRYPTED column indicates whether a tablespace is encrypted

限制

The following list includes the restrictions that apply to TDE tablespace encryption:

External Large Objects (BFILEs) cannot be encrypted using TDE tablespace encryption. This is because these files reside outside the database.

To perform import and export operations, use Oracle Data Pump.

注意：You cannot access any encrypted data without the master encryption key

》管理TDE

1 启用AUTO LOGIN wallet

2 备份和管理主密钥

如果没有主密钥是不能访问加密数据的，主密钥存储在wallet中，需要对wallet做备份

当产生新的主密钥时必须备份wallet

备份时不应同时备份加密数据和钱包wallet，应单独备份wallet，以防止恶意用户同时获取加密数据和wallet钱包

rman不会备份wallet

rman如果使用了Oracle Secure Backup OSB（如带库）会自动排除auto login wallet（cwallet.sso），但不会排除普通的wallet（ewallet.p12 ），所以使用此备份方式，建议添加以下语句排除wallet

exclude name \*.p12

如果存储主密钥的wallet丢失或损坏，将备份版本拷贝至适当位置即可恢复

1 如果备份版本是上次重置主密钥之后版本，无需任何操作

2 如果备份版本是上次重置主密钥之前版本，即不包含最新主密钥，则需要将数据库恢复至重置主密钥之前时间点。之后的数据将丢失。

================================

》其他加密

加密数据无法使用exp/imp工具移动，只能使用datapump移动数据，

导出TDE加密 数据时，必须打开钱包wallet，因为数据需要主密钥解密

使用数据泵datapump或恢复管理器rman移动数据时，如果使用TDE加密功能，需要wallet。也可以使用工具自带的加密功能单独加密数据。

数据泵（datapump）加密

1 涉及TDE数据时，需要wallet钱包打开，只适合本机原地操作。

导出数据后，数据以非加密的形式存储在转储文件集中

expdp hr/hr TABLES=emp DIRECTORY=DATA\_PUMP\_DIR DUMPFILE=emp.dmp logfile=emp.expdp

ORA-39173: Encrypted data has been stored unencrypted in dump file set.

ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY "oracle123";

impdp \'/ as sysdba\' DIRECTORY=DATA\_PUMP\_DIR DUMPFILE=emp.dmp logfile=emp.impdp remap\_schema=hr:test

如果想转储文件集也加密，可以单独指定以下参数（TDE数据解密后再加密）

ENCRYPTION\_PASSWORD

ENCRYPTION = [ALL | DATA\_ONLY | ENCRYPTED\_COLUMNS\_ONLY | METADATA\_ONLY | NONE]

ALL enables encryption for all data and metadata in the export operation.

DATA\_ONLY specifies that only data is written to the dump file set in encrypted format.

ENCRYPTED\_COLUMNS\_ONLY specifies that only encrypted columns are written to the dump file set in encrypted format. To use this option, you must have Oracle Advanced Security transparent data encryption enabled. See Oracle Database Advanced Security Administrator's Guide for more information about transparent data encryption.

METADATA\_ONLY specifies that only metadata is written to the dump file set in encrypted format.

NONE specifies that no data is written to the dump file set in encrypted format.

ENCRYPTION不指定默认为all

2 不涉及TDE数据时，不需要wallet钱包，可单独加密转储文件集。适用于所有场景

expdp hr/hr TABLES=employees DIRECTORY=DATA\_PUMP\_DIR DUMPFILE=employees.dmp logfile=employees.expdp encryption\_password=oracle

impdp \'/ as sysdba\' DIRECTORY=DATA\_PUMP\_DIR DUMPFILE=employees.dmp logfile=employees.impdp remap\_schema=hr:test

ORA-39174: Encryption password must be supplied.

encryption\_password=oracle

RMAN备份加密

1 使用TDE，适合本机原地操作

1）创建钱包（已创建）

2）打开钱包

SQL> alter system set encryption wallet open identified by oracle;

3）开启RMAN加密功能

RMAN> configure encryption for database on;

4）备份与恢复命令不会发生更改，需要打开钱包

5）通过命令可以暂时覆盖永久配置

RMAN> set encryption off;

2 口令模式设置（不需要钱包），使用与所有场景

RMAN> set encryption on identified by admin only;

RMAN> backup datafile 4;

RMAN> set decryption identified by admin;    --还原时必须解密

RMAN> restore datafile 4;

3 如果创建了钱包，可以使用双重模式

SQL> alter system set encryption wallet open identified by oracle;

RMAN> set encryption on identified by oracle;

RMAN> backup datafile 4;

SQL> alter system set encryption wallet close;   --关闭钱包

RMAN> set decryption identified by oracle;    --钱包、口令任选一个 SQL> alter system set encryption wallet open identified by oracle;

RMAN> restore datafile 4;

RMAN> recover datafile 4;

RMAN> alter database open;

改变rman加密算法

SQL> select algorithm\_name from v$rman\_encryption\_algorithms;

RMAN> configure encryption algorithm 'AES192';

RMAN> set encryption algorithm 'AES256';

relate view

在dba\_tablespaces会添加新列encrypted让你查看状态

v$encryption\_wallet

v$encrypted\_tablespaces